



## UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
[www.uspto.gov](http://www.uspto.gov)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/917,493	07/27/2001	Daniel Cook Jarvis	10010790-1	9794
7590 HEWLETT-PACKARD COMPANY Intellectual Property Administration P. O. Box 272400 Fort Collins, CO 80527-2400			EXAMINER	
			MILIA, MARK R	
			ART UNIT	PAPER NUMBER
			2625	
			MAIL DATE	DELIVERY MODE
			05/28/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE BOARD OF PATENT APPEALS  
AND INTERFERENCES

---

*Ex parte* DANIEL C. JARVIS and DANIEL J. DYER

---

Appeal 2008-005847  
Application 09/917,493  
Technology Center 2600

---

Decided:<sup>1</sup> May 28, 2009

---

Before KENNETH W. HAIRSTON, JOHN A. JEFFERY,  
and BRADLEY W. BAUMEISTER, *Administrative Patent Judges*.

BAUMEISTER, *Administrative Patent Judge*.

DECISION ON APPEAL

---

<sup>1</sup> The two-month time period for filing an appeal or commencing a civil action, as recited in 37 C.F.R. 1.304, begins to run from the decided date shown on this page of the decision. The time period does not run from the Mail Date (paper delivery) or Notification Date (electronic delivery).

## STATEMENT OF CASE

Appellants appeal under 35 U.S.C. § 134 (2002) from the Examiner's rejection of claims 1-37. We have jurisdiction under 35 U.S.C. § 6(b) (2002). We affirm.

### A. *Appellants' invention*

Appellants' invention relates to a manager loadable printer and methods of its use (App. Br. 3). The printer comprises an application program loaded on the printer, wherein a manager invokes functionality on and receives results from the application program via an agent remotely located from the application program (*Id.*).

### B. *The claims*

Independent claim 1 is illustrative.<sup>2</sup> It reads as follows:

1. A manager loadable printer comprising:

---

<sup>2</sup> Appellants nominally argue dependent claims 2-5 separately from independent claim 1 (App. Br. 9); dependent claims 7-16 separately from independent claim 6 (App. Br. 12); dependent claims 18, 19 and 22-24 separately from independent claim 17 (App. Br. 15); dependent claims 26-30 and independent claim 31 separately from independent claim 25 (App. Br. 18); and dependent claims 35-37 separately from independent claim 32 (App. Br. 21). However, the arguments presented for claims 2-5; 7-16; 18, 19 and 22-24; 26-30; and 35-37 are substantially the same as those presented for claims 1, 6, 17, 25, and 32. Accordingly, we group the following claims together: (1) claims 1-5; (2) claims 6-16; (3) claims 17-19 and 22-24; (4) claims 25-30; and (5) claims 32 and 35-37. We also select claims 1, 6, 17, 25, and 32 as representative of these respective groups. *See* 37 C.F.R. § 41.37(c)(1)(vii).

an application program loaded on the printer, wherein a manager invokes functionality on and receives results from the application program via an agent remotely located from the application program.

*C. The references and rejections*

The Examiner relies on the following prior art references to show unpatentability:

Yan	US 6,003,065	Dec. 14, 1999
Friedman	US 6,763,499 B1	July 13, 2004 (filed July 26, 1999)
Sokolov	US 6,823,504 B1	Nov. 23, 2004 (filed Nov. 15, 2000)

1. Claims 1-19, 22-32, and 35-37 stand rejected under 35 U.S.C. § 102(b) as anticipated by Yan.
2. Claims 20 and 33 stand rejected under 35 U.S.C. § 103(a) as obvious over Yan in view of Sokolov.
3. Claims 21 and 34 stand rejected under 35 U.S.C. §103(a) as obvious over Yan in view of Friedman.

Rather than repeat the arguments of the Appellants or the Examiner, we refer to the Brief and the Answer for their respective details.<sup>3</sup> In this decision, we have considered only those arguments actually made by Appellants. Arguments which Appellants could have made but did not make

---

<sup>3</sup> We refer to (1) the Appeal Brief filed Dec. 15, 2006; (2) the Examiner's Answer mailed Apr. 3, 2007; (3) the Reply Brief filed June 4, 2007; (4) the Supplemental Examiner's Answer mailed Nov. 15, 2007; and (5) the Supplemental Reply Brief filed Jan. 15, 2008 throughout this opinion.

in the Brief have not been considered and are deemed to be waived. *See* 37 C.F.R. § 41.37(c)(1)(vii).

#### ISSUE

The Examiner asserts that the various claimed elements (*see e.g.*, claim 1) read on the following elements of Yan (*see e.g.*, Yan, fig. 2): the “application program loaded on the printer” reads on Yan’s executable computer program 226; the claimed manager reads on Yan’s virtual machine (VM) instruction processor 214; and the claimed agent reads on Yan’s peripheral application programming interface (API) 228 (Ans. 4).

Appellants assert that (1) Yan’s “virtual machine instructions processor does not invoke functionality on an application program loaded on a printer” (Reply Br. 2) and (2) “*Yan* fails to teach or suggest that a manager from a manager loadable printer invokes functionality on an application program via an agent remotely located from the application program” (App. Br. 8). Appellants also assert that because each of Yan’s peripheral devices has its own API, Yan does not disclose than an agent is remotely located from an application program (Supp. Reply Br. 3).

The issues before us, then, are:

- I. Have Appellants shown that the Examiner erred in interpreting the claimed “application program loaded on the printer” as reading on Yan’s VM instruction processor 214 and the claimed “manager” as reading on Yan’s executable computer program 226?
- II. Regarding claim 1, have Appellants shown that the Examiner erred in finding that Yan discloses an application program loaded on a printer wherein the application program is capable of receiving functionality from,

and sending results to, a manager via an agent that is remotely located from the application program?

III. Regarding claim 6, have Appellants shown that the Examiner erred in finding that Yan discloses a method of instructing a printer having a virtual machine, the method comprising: providing an agent, the agent having an associated applet; loading the applet on the virtual machine; and executing the applet on the virtual machine, wherein a manager invokes functionality on and receives results from the applet via the agent remotely located from the applet?

IV. Regarding claims 17 and 32, have Appellants shown that the Examiner erred in finding that Yan discloses an applet; a VM capable of executing the applet; and an interface that is capable of communication between the printer and a remote agent, wherein the agent initiates management events including requesting amount of resources being utilized by each applet operating on the virtual machine?

V. Regarding claim 25, have Appellants shown that the Examiner erred in finding that Yan discloses a method of instructing a printer having a virtual machine comprising: serving an applet to the printer; executing the applet on the virtual machine to produce a result; communicating the result from the printer to an agent remotely located from the printer; and communicating the result from the agent to a manager?

#### FINDINGS OF FACT

The record supports the following Findings of Fact (FF) by a preponderance of the evidence:

1. Yan discloses a printer processing unit 106B that includes primary storage 216, VM instruction processor 214, and input-output (I/O) interface 224 that facilitates communication between the primary storage and the processor (Yan, fig. 2; col. 7, ll. 49-58).
2. Yan's VM Instruction processor 214 implements the JAVA virtual machine (Yan, col. 8, ll. 46-49).
3. The primary storage of Yan's printer processing unit includes an executable computer program 226, a peripheral API 228 and optionally an operating system 230 for managing resources on the peripheral printer or host computer (Yan, fig. 2; col. 9, ll. 26-30).
4. Yan's peripheral API:  
enables executable computer programs 226 to access functionality associated with a peripheral device such as printer 102B using hardware independent and architecturally neutral system calls. These system calls correspond to specific virtual machine instructions which execute on virtual machine instruction processor 214 in the form of bytecodes and cause the peripheral device to operate in a specified manner.  
(Yan, col. 9, ll. 33-39).
5. Appellants do not dispute that "an agent" reads on Yan's API 228.
6. The Specification states, "[w]hen a manager 510, 514 and an agent 530, 534 reside on different computers, referred to herein as a "remote" manager, the interface optionally includes connectors and/or adaptors (Spec., 11).
7. The Specification states, "[a] resident manager normally resides on a device having a manageable agent thereon. A remote manager normally

resides on a device separate from and in communication with the device on which the agent resides" (Spec. 13).

8. Yan further discloses the API is implemented with a VM instruction set, such as an applet (Yan, col. 10, ll. 35-44).

9. Yan's applet is downloaded into the VM instruction processor (Yan, col. 10, ll. 24-44).

10. Yan discloses that the printer processing unit 106B includes a primary network interface 212 that couples the printer to networks, routers, host computers and other peripheral devices (Yan, col. 8, ll. 9-21; fig. 2).

11. Yan's printer processing unit 106B includes I/O interface 224 which facilitates communication between the VM instruction processor 214 and the primary network interface (Yan, col. 7, ll. 49-58; fig. 2).

12. Yan discloses:

In operation, the remote administration application downloads virtual machine instruction applet over a network connection into the peripheral device which requests the peripheral device's control applet. Upon execution in the peripheral device, the downloaded virtual machine instruction applet then instructs the peripheral device to locate and return the control applet within the peripheral device which understands the operation of the device. The control applet is then plugged into the skeletal remote administration application and enables a person to manage the particular peripheral device.

(Yan, col. 23, ll. 27-37).

#### PRINCIPLES OF LAW

1. "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior

art reference.” *Verdegaal Bros., Inc. v. Union Oil Co. of Cal.*, 814 F.2d 628, 631 (Fed. Cir. 1987).

2. Appellant has the burden on appeal to the Board to demonstrate error in the Examiner’s position. *See In re Kahn*, 441 F.3d 977, 985-86 (Fed. Cir. 2006).

3. During examination, the claims must be interpreted as broadly as their terms reasonably allow. *In re Am. Acad. of Sci. Tech. Ctr.*, 367 F.3d 1359, 1369 (Fed. Cir. 2004).

4. A claim containing a “recitation with respect to the manner in which a claimed apparatus is intended to be employed does not differentiate the claimed apparatus from a prior art apparatus” if the prior art apparatus teaches all the structural limitations of the claim. *Ex parte Masham*, 2 USPQ2d 1647 (BPAI 1987) (non-precedential).

## ANALYSIS

### I.

Appellants dispute the Examiner’s contention that “an application program loaded on the printer” and “a manager” (*see e.g.*, claim 1) respectively read on Yan’s VM instruction processor 214 and executable computer program 226 (Ans. 4, 12-13; App. Br. 7-9; Reply Br. 2-4; Supp. Ans. 3; Supp. Reply Br. 2-4). We agree with Appellants.

Yan discloses a printer processing unit 106B that includes primary storage 216, VM instruction processor 214, and input-output (I/O) interface 224 that facilitates communication between the primary storage and the processor (FF 1). The VM Instruction processor 214 implements the JAVA

virtual machine (FF 2). That is, VM instruction processor of Yan's printer is loaded with an application program.

The primary storage of Yan's printer processing unit includes an executable computer program 226, a peripheral API 228, and optionally an operating system 230 for managing resources on the peripheral printer or host computer (FF 3). Yan's peripheral API:

*enables executable computer programs 226 to access functionality associated with a peripheral device such as printer 102B using hardware independent and architecturally neutral system calls. These system calls correspond to specific virtual machine instructions which execute on virtual machine instruction processor 214 in the form of bytecodes and cause the peripheral device to operate in a specified manner.*

(FF 4 (emphasis added)).

This passage considered with the disclosure of Yan as a whole, indicates that the Examiner's interpretation is backwards. One of ordinary skill in the art would reasonably understand that the executable computer program 226 invokes functionality on and receives results from the application program loaded on the VM instruction processor 214 via the API 228. That is, the claimed "manager" reads on Yan's executable computer program 226—either alone or alternatively in combination with the operating system 230;<sup>4</sup> the claimed "agent" reads on Yan's peripheral API 228; and the "application program loaded on the printer" reads on the VM software loaded on Yan's VM instruction processor 214.

---

<sup>4</sup> The stated options of (1) Yan's executable computer program 226 alone and (2) Yan's executable computer program 226 in combination with the operating system 230 are hereinafter more simply referred to as the executable computer program 226 for brevity.

The fact that the Examiner misinterpreted Yan's teachings is not dispositive, however, regarding whether Yan anticipates the appealed claims. Accordingly, we address each of the independent claims separately.

## II.

Independent claim 1 recites:

1. A manager loadable printer comprising:

an application program loaded on the printer, *wherein a manager invokes functionality on and receives results from the application program via an agent remotely located from the application program.*

(emphasis added).

As noted above in Section I, the claimed manager, agent, and application program respectively read on Yan's executable computer program 226, peripheral API 228 and VM software loaded on VM instruction processor 214. Further, Appellants do not dispute that "an agent" reads on Yan's API 228 (FF 5). Rather, Appellant's argue that Yan's API is not "remotely located from an application program" (Supp. Reply Br. 3). The only question then is whether Yan's API or "agent" may be interpreted as being "remotely located from the application program," as required by claim 1.

Appellants' Specification first states that the term "remote manager" is used to indicate that the manager resides on a different computer than the agent (FF 6). But the Specification subsequently states that, "[a] *resident* manager *normally* resides on a device having a manageable agent thereon,"

and “[a] *remote* manager *normally* resides on a device separate from and in communication with the device on which the agent resides” (FF 7 (emphasis added)). As such, the Specification initially appears to indicate that two elements (e.g., an agent and a manager) must reside on separate devices (e.g., computers) in order for one of the elements to be deemed to be “remote.” However, Appellants’ use of the word “normally” more specifically indicates that two elements do not *necessarily* have to reside on separate devices in order for one element to be characterized as “remote.” That is, a first element may be characterized as being “remote” from a second element even if they are located on the same device.

Based upon Appellants’ cryptic usage of the term “remote,” we understand claim 1 to require that the agent be in a different location from the application program,<sup>5</sup> albeit both potentially located within the printer. Yan discloses that the API (“agent”) is located within primary storage 216 (FF 3), and that the VM (“application program”) is located on the VM instruction processor (FF 2). As the primary storage and VM instruction processor are separate and discrete components (*see e.g.*, fig. 2), the API may be reasonably viewed as being remote from the VM.

Furthermore, even if the term “remote” were to require a narrower interpretation (e.g., indicating that the agent and application program be located in different devices), Yan would still anticipate claim 1. The only elements positively recited in claim 1 are the printer and the application program. Neither the manager nor the agent is positively recited. Rather,

---

<sup>5</sup> The Free Dictionary; <http://encyclopedia2.thefreedictionary.com/remote> (last accessed May 17, 2009) (defining “remote” as any entity (user, machine, facility) in a different geographical location).

the italicized language of claim 1 is directed towards the functionality, or intended use, of the application program. As such, Yan would anticipate claim 1 so long as it merely discloses an application program loaded on a printer wherein the application program *is capable of* receiving functionality from, and sending results to, a manager via an agent that is remotely located from the application program, as required by claim 1. *See Masham*, 2 USPQ2d at 1647.

Yan's VM instruction processor receives functionality from, and sends results to, the manager and agent of the primary storage via the I/O interface (FF 1-4, 11). As such, Yan's VM ("application program") possesses the functionality of claim 1 and is therefore capable of communicating with a manager/agent pair that was alternatively located separate from the printer.

For the foregoing reasons, Appellants have not persuaded us of error in the Examiner's anticipation rejection of representative claim 1. Accordingly, we will sustain the Examiner's rejection of that claim and dependent claims 2-5 which fall with claim 1.

### III.

Independent method claim 6 is narrower than independent product claim 1. Claim 6 positively recites the manager and agent. Claim 6 further requires that (1) the manager invokes functionality on and receives results from an applet via the agent remotely located from the applet; and (2) the applet is loaded into the VM from the agent.

Yan further discloses the API (or “agent”) is implemented with a VM instruction set, such as an applet (FF 8). The applet is downloaded into the VM instruction processor (FF 9). Applying the broader interpretation of “remote” that was set forth in Section I above, after the applet is downloaded from the API to the VM instruction processor, the applet may be deemed to be “remote” from the API. This is so because the memory in which the AI resides is distinct or separate from the VM instruction processor (*see fig. 2*).

For the foregoing reasons, Appellants have not persuaded us of error in the Examiner’s anticipation rejection of representative claim 6. Accordingly, we will sustain the Examiner’s rejection of that claim and dependent claims 7-16 which fall with claim 6.

#### IV.

Independent claim 17 recites:

17. A printer comprising:
  - an applet;
  - a virtual machine capable of executing the applet; and

*an interface for communication between the printer and a remote agent, wherein the agent initiates management events including requesting amount of resources being utilized by each applet operating on the virtual machine.*

(emphasis added). The only elements positively recited in claim 17 are the printer, the applet, the virtual machine, and the interface. The remote agent initiating management events is not positively recited. Rather, the italicized language of claim 17 is directed towards the functionality, or intended use,

of the interface. As such, Yan would anticipate claim 17 so long as it merely discloses an applet; a VM capable of executing the applet; and an interface that *is capable of* communication between the printer and a remote agent, wherein the agent initiates management events including requesting amount of resources being utilized by each applet operating on the virtual machine. *See Masham*, 2 USPQ2d at 1647.

As was already addressed above, Yan discloses an applet that is downloaded to, and executed by a VM (FF 9). Yan also discloses that the printer processing unit 106B includes a primary network interface 212 that couples the printer to networks, routers, host computers and other peripheral devices (FF 10). The printer processing unit 106B also includes I/O interface 224 which facilitates communication between the VM instruction processor 214 and the primary network interface (FF 11). As such, the I/O interface—either alone or in combination with the primary network interface—is capable of communication between the printer and a remote agent, wherein the agent initiates management events including requesting amount of resources being utilized by each applet operating on the virtual machine.

For the foregoing reasons, Appellants have not persuaded us of error in the Examiner's anticipation rejection of representative claim 17. Accordingly, we will sustain the Examiner's rejection of that claim and dependent claims 18, 19 and 22-24 which fall with claim 17.

Independent claim 32 is similar in scope to claim 17. Instead of reciting "*a virtual machine* capable of executing the applet" as in claim 17 though, claim 32 recites "*an applet execution means* for executing the

applet.” As “a virtual machine capable of executing an applet” is a specific example of “an applet execution means for executing an applet,” claim 32 is broader in scope than claim 17. Accordingly, we will also sustain the Examiner’s rejection of claim 32 and dependent claims 35-37 which fall with claim 32.

Likewise, we will sustain the Examiner’s obviousness rejections of claims 20 and 33 over Yan in view of Sokolov and claims 21 and 34 over Yan in view of Friedman. Appellants have not particularly pointed out errors in the Examiner’s reasoning to persuasively rebut the Examiner’s *prima facie* case of obviousness, but merely reiterate the same arguments with respect to the alleged deficiencies of Yan in connection with independent claims 17 and 32 (App. Br. 21-24). We are not persuaded by these arguments, however, for the same reasons discussed above.

V.

Independent method claim 25 recites:

25. A method of instructing a printer having a virtual machine comprising:

serving an applet to the printer;

executing the applet on the virtual machine to produce a result;

communicating the result from the printer to an agent remotely located from the printer; and

communicating the result from the agent to a manager.

The Examiner's rejection of claim 25 cites to various passages of Yan contained in columns 22-24 relating to Yan's protocol for distributed processing on host and peripheral device (*see* Yan, col. 22, l. 11-col. 24, l. 67). Most relevant portion cited is:

In operation, the remote administration application downloads virtual machine instruction applet over a network connection into the peripheral device which requests the peripheral device's control applet. Upon execution in the peripheral device, the downloaded virtual machine instruction applet then instructs the peripheral device to locate and return the control applet within the peripheral device which understands the operation of the device. The control applet is then plugged into the skeletal remote administration application and enables a person to manage the particular peripheral device.

(FF 12).

The Examiner's rationale for the rejection (Ans. 5) is unquestionably less than ideal; the rejection does not set forth which particular elements of the host computer are respectively interpreted as specifically constituting the agent and the manager. Furthermore, the passages of Yan cited in the rejection are lengthy and detailed in describing how the protocol is carried out. We are not fully certain how the Examiner is interpreting Yan.

Nonetheless, we understand the Examiner's position to be as follows. "The remote administration application downloads virtual machine instruction applet over a network connection into the peripheral device which requests the peripheral device's control applet" (FF 12). The downloading of this applet constitutes "serving an applet to the printer" per claim 25. The downloaded VM instruction applet instructs the peripheral device (or printer) to locate a control applet and return it to the host's

skeletal remote administration application (FF 12). The control applet constitutes a “result from the printer;” the host’s skeletal remote administration application constitutes a “remote agent.” Accordingly, the returning of the applet to the skeletal remote administration application constitutes “communicating the result from the printer to an agent remotely located from the printer,” per claim 25.

Furthermore, the “person [managing] the particular peripheral device” (FF 12) may be deemed to constitute a “manager.” In order for this manager to be aware that the control applet has been successfully loaded into the skeletal remote administration application, the application will have to cause some indication of this event to be sent to the manager (for example, cause a visual indication to be displayed on a network video monitor). *See In re Preda*, 401 F.2d 825, 826 (CCPA 1968) (noting that “in considering the disclosure of a reference, it is proper to take into account not only specific teachings of the reference but also the inferences which one skilled in the art would reasonably be expected to draw therefrom”). As such, the skeletal remote administration application’s act of indicating to the manager that the skeletal remote administration application (“agent”) has successfully received the control applet (“result”) constitutes “communicating the result from the agent to a manager,” per claim 25.

Appellants do not dispute that Yan discloses a method that includes serving an applet to a printer, executing the applet on a VM to produce a result, and communicating the result from printer to a host computer (*see App. Br. 15-18*). Appellants even acknowledge, “Yan discloses that a control applet is obtained from a peripheral device and is plugged into the

remote administration application ‘and enables a person to manage the particular peripheral device.’ Col. 23, lines 30-35” (App. Br. 17). Rather, Appellants argue that “Yan fails to teach or suggest at least ‘communicating the result from the printer to an agent remotely located from the printer; and communicating the result from the agent to a manager’” (App. Br. 16). More specifically, Appellants only argue that (1) the host computer’s remote administration application sans control applet cannot constitute a manager, and (2) the remote administration application having a plugged-in control applet may possibly constitute a manager, but no other element is available to constitute the claimed agent (App. Br. 16-17).

Appellants’ arguments, however, do not address why the claimed “agent” may not be reasonably interpreted as reading on Yan’s skeletal remote administration applet. Nor do Appellants’ arguments address why the claimed “manager” cannot be reasonably interpreted as reading on a human being. As such, Appellants’ have not met their burden in showing why the Examiner’s anticipation rejection of representative claim 25 over Yan was improper. Accordingly, we will sustain the Examiner’s rejection of claim 25 as well as the rejection of dependent claims 26-30 and independent claim 31 which fall with claim 25.

#### CONCLUSION OF LAW

I. Appellants have not shown that the Examiner erred in interpreting the claimed “application program loaded on the printer” as reading on Yan’s VM instruction processor 214 and the claimed “manager” as reading on Yan’s executable computer program 226.

II. Regarding claim 1, Appellants have not shown that the Examiner erred in finding that Yan discloses an application program loaded on a printer wherein the application program is capable of receiving functionality from, and sending results to, a manager via an agent that is remotely located from the application program.

III. Regarding claim 6, Appellants have not shown that the Examiner erred in finding that Yan discloses a method of instructing a printer having a virtual machine, the method comprising: providing an agent, the agent having an associated applet; loading the applet on the virtual machine; and executing the applet on the virtual machine, wherein a manager invokes functionality on and receives results from the applet via the agent remotely located from the applet.

IV. Regarding claims 17 and 32, Appellants have not shown that the Examiner erred in finding that Yan discloses an applet; a VM capable of executing the applet; and an interface that is capable of communication between the printer and a remote agent, wherein the agent initiates management events including requesting amount of resources being utilized by each applet operating on the virtual machine.

V. Regarding claim 25, Appellants have not shown that the Examiner erred in finding that Yan discloses a method of instructing a printer having a virtual machine comprising: serving an applet to the printer; executing the applet on the virtual machine to produce a result; communicating the result from the printer to an agent remotely located from the printer; and communicating the result from the agent to a manager.

As such, Appellants have not shown that the Examiner erred in rejecting claims 1-37 under §§ 102 and 103.

#### DECISION

We sustain the Examiner's rejections with respect to all pending claims on appeal. Therefore, the Examiner's rejection of claims 1-37 is affirmed.

No time period for taking any subsequent action in connection with this appeal may be extended under 37 C.F.R. § 1.136(a)(1)(iv).

AFFIRMED

KIS

HEWLETT-PACKARD COMPANY  
Intellectual Property Administration  
P. O. Box 272400  
Fort Collins, CO 80527-2400